

**PEMBUATAN SISTEM INFORMASI PEMBAYARAN BERBASIS WEB
PONDOK PESANTREN ISLAM AL-MUKMIN (PPIM)**



**Disusun sebagai salah satu syarat memperoleh Gelar Strata I
pada Jurusan Informatika Fakultas Komunikasi dan Informatika**

**Oleh:
ABIQS FASTABIQULKHOIROT
L200150055**

**PROGRAM STUDI INFORMATIKA
FAKULTAS KOMUNIKASI DAN INFORMATIKA
UNIVERSITAS MUHAMMADIYAH SURAKARTA
2021**

HALAMAN PERSETUJUAN

**PEMBUTAN SISTEM INFORMASI PEMBAYARAN BERBASIS WEB
PONDOK PESANTREN AL-MUKMIN (PPIM)**


PUBLIKASI ILMIAH

oleh:

ABIQS FASTABIQULKHOIROT
L200150055

Telah diperiksa dan disetujui untuk diuji oleh:

Dosen Pembimbing


Heru Supriyono, S.T., M.Sc., PhD
NIK.970

HALAMAN PENGESAHAN

**PEMBUTAN SISTEM INFORMASI PEMBAYARAN BERBASIS WEB
PONDOK PESANTREN AL-MUKMIN (PPIM)**

OLEH

ABIQS FASTABIQULKHOIROT

L200150055

**Telah dipertahankan di depan Dewan Penguji
Pada hari Rabu, 30 Juni 2021
dan dinyatakan telah memenuhi syarat**

Dewan Penguji:

1. Heru Supriyono, S.T., M.Sc., PhD

(Ketua Dewan Penguji)

(.....)

2. Dr. Endah Sudarmilah, M.Eng

(Anggota I Dewan Penguji)

(.....)

3. Aris Rakhmadi, S.T., M.Eng

(Anggota II Dewan Penguji)

(.....)

**Dekan
Fakultas Komunikasi dan Informatika**



**Nurgiyatna, S.T., M.Sc., PhD
NIK.881**

PERNYATAAN

Dengan ini saya menyatakan bahwa dalam publikasi ilmiah ini tidak terdapat karya yang pernah diajukan untuk memperoleh gelar kesarjanaan di suatu perguruan tinggi dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan orang lain, kecuali secara tertulis diacu dalam naskah dan disebutkan dalam daftar pustaka.

Apabila kelak terbukti ada ketidakbenaran dalam pernyataan saya di atas, maka akan saya pertanggungjawabkan sepenuhnya.

Surakarta, 30 Juni 2021

Penulis



ABIQS FASTABIOULKHOIROT
L200150055

PEMBUTAN SISTEM INFORMASI PEMBAYARAN BERBASIS WEB PONDOK PESANTREN AL-MUKMIN (PPIM)

Abstrak

Pembayaran SPP adalah suatu transaksi pada suatu sistem persekolahan. Hal ini bersifat penting karena berkaitan dengan administrasi sekolah. Pembayaran SPP terkadang mengalami kendala pada saat pendataan serta proses informasi pembayaran yang dilakukan selama ini masih menggunakan buku besar sehingga membuat pekerjaan menghabiskan banyak waktu dan tenaga. Perkembangan teknologi pada era sekarang telah meningkat seiring perkembangan zaman. Sistem pembayaran pada Pondok Pesantren Al-mukmin masih menggunakan cara manual, dimana petugas bendahara masih menulis data pembayaran, kuitansi, dan laporan kepada Yayasan Pendidikan Islam dan Asuhan Yatim oleh karena itu sering terjadi masalah dalam melakukan pembayaran. Tujuan sistem ini dibuat untuk menghasilkan sistem pembayaran yang berbasis komputer yang mempunyai fitur pelayanan pembayaran dan pembuatan laporan pembayaran. Metode yang digunakan adalah metode *System Development Life Cycle* (SDLC) jenis *Waterfall*. Aplikasi ini membantu untuk pembukuan dan pencetakan dalam pembayaran. Hasil dari pembuatan sistem ini adalah sistem informasi untuk bagian bendahara sehingga dapat dengan mudah dan cepat dalam melakukan pembayaran. Menurut pengujian yang telah dilakukan dengan metode *Black-box* sistem dapat berjalan dengan baik dan sesuai dengan perancangannya dalam melakukan pelayanan pembayaran, mencetak struk, dan membuat laporan.

Kata Kunci: *Pembayaran spp, Sistem Informasi, Waterfall.*

Abstract

SPP payment is a transaction in a school system. This is important because it relates to school administration. SPP payments sometimes experience problems when collecting data and processing payment information so far they are still using a ledger so that it takes up a lot of time and effort. The development of technology in the present era has increased along with the times. The payment system at the Al-Mukmin Islamic Boarding School still uses the manual method, where treasury officers still write payment data, receipts, and reports to the Islamic Education Foundation and Orphanage, therefore problems often occur in making payments. The purpose of this system is to produce a computer-based payment system that features payment services and the creation of payment reports. The method used is the Waterfall type System Development Life Cycle (SDLC) method. This application helps for bookkeeping and printing in payments. The result of making this system is an information system for the treasurer so that it can easily and quickly make payments. According to the tests that have been done with the Black-box method, the system can run well and in accordance with the design in making payment services, printing receipts, and making reports

Keywords: *Payment of spp, Information Systems, Waterfall.*

1. PENDAHULUAN

Perkembangan dalam bidang ilmu pengetahuan dan teknologi dikembangkan dengan tujuan untuk memberikan kemudahan bagi manusia dalam melaksanakan tugas dan kepentingannya. Banyak sekali bagian ilmu pengetahuan dan teknologi yang mengalami perkembangan yang begitu pesatnya, hal itu didukung oleh sumber daya alam dan sumber daya manusia yang memadai dan berkualitas. Salah satu diantaranya adalah bidang teknologi informasi dan pengolahan data. Saat ini suatu bentuk informasi dan data bisa dibuat sesuai dengan apa yang kita inginkan, dan banyak peluang yang bisa dimanfaatkan untuk mengembangkannya.

Pondok Pesantren Al-mukmin merupakan pesantren di Ngruki, Solo yang didirikan oleh “enam serangkai” antara lain adalah Abdullah Sungkar, Abu Bakar Ba’asyir, Yoyok Roswadi, Abdullah Baradja, Abdul Qohar H. Daeng Matase, dan Hasan Basri pada tahun 1972. Tenaga kependidikan di Pondok Pesantren Al-mukmin memiliki sekitar 236 orang Guru dan Karyawan, yang mendidik santri berjumlah kurang lebih 663 santri putra dan 814 santri putri.

Menurut David, Tanui, dan Oruta (2019) Sistem manajemen membantu dalam manajemen keuangan melalui sistem informasi yang disesuaikan untuk meningkatkan transaksi keuangan di sekolah. Manajemen keuangan merupakan tindakan pengurusan/ketatausahaan keuangan yang meliputi pencatatan, perencanaan, pelaksanaan, pertanggungjawaban dan pelaporan. Dengan demikian, manajemen keuangan sekolah dapat diartikan sebagai rangkaian aktivitas mengatur keuangan sekolah mulai dari perencanaan, pembukuan, pembelanjaan, pengawasan dan pertanggung-jawaban keuangan sekolah. Pembiayaan pendidikan hendaknya dilakukan secara efisien.

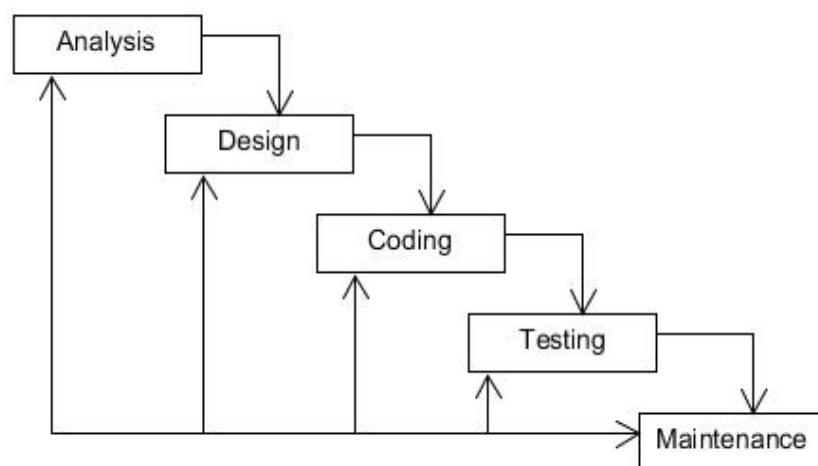
Menurut Susanto (2018) Permasalahan muncul ketika bendahara melakukan rekapan data siswa yang sudah membayar dan yang belum membayar SPP dan iuran siswa, hal ini menyebabkan banyaknya waktu yang terbuang dan buku catatan pembayaran akan menjadi rusak sehingga pelaporan pembayaran perbulan menjadi terlambat. Oleh sebab itu, tentu adanya sebuah sistem yang diberlakukan untuk kelancaran transaksi tersebut. Sistem tersebut mencakup bagaimana proses transaksi itu terjadi. Untuk memudahkan pengolahan data pembayaran yang terjadi hingga pembuatan laporan, sangat dibutuhkan sistem informasi pembayaran tersebut.

Berdasarkan informasi yang didapat, diketahui bahwa pengelolaan pembayaran Pondok Pesantren Al-mukmin masih dengan cara manual, sehingga menyebabkan kesulitan dalam pencarian dan menghabiskan tenaga dalam penulisan kuitansi dan laporan pembayaran yang sudah dilakukan. Bagian bendahara atau *syahriah* hanya berjumlah 4 orang, 2 orang untuk mengurus santri putra dan 2 orang untuk santri putri. Pembayaran yang dilakukan hanya ditulis dalam buku besar sehingga membutuhkan banyak waktu dan menghabiskan banyak tenaga dalam pencarian dan penulisan.

Solusi yang ditawarkan untuk masalah yang dihadapi oleh pesantren dapat dikembangkan dalam sistem pembayaran dengan pembuatan sistem informasi pembayaran. Sistem informasi ini dapat melakukan penambahan pembayaran pada santri sekaligus mencetak kuitansi, mencetak laporan yang akan dikirim pada yayasan, pencarian pembayaran pada santri sehingga lebih efektif dan efisien.

2. METODE

Metode pengembangan sistem yang digunakan dalam pembuatan sistem informasi ini adalah metode SDLC model *waterfall*. Alur proses dibagi menjadi beberapa fase SDLC seperti analisis kebutuhan, perencanaan dan penjadwalan, desain sistem, implementasi, pengujian, penyebaran dan pemeliharaan (Bhavsar, Shah, & Gopalan, 2020). Model *waterfall* dapat dilihat pada Gambar 1.



Gambar 1. Tahapan metode *waterfall* menurut Bhavsar, Shah, dan Gopalan (2020)

2.1 Analisis

Analisis Sistem melibatkan kedua kelayakan studi dan analisis sistem kerangka SDLC. Tim pengembangan menganalisis kelayakan dan fitur produk perangkat lunak berdasarkan persyaratan pengguna (Gharajeh, 2019). Proses analisa kebutuhan dilakukan dengan cara wawancara atau bertanya kepada bagian pembayaran yang terkait dalam sistem tersebut agar mengetahui tentang kebutuhan yang diperlukan untuk pembuatan system. Berikut adalah kebutuhan yang diperlukan dalam pembuatan sistem dan kebutuhan dalam pengoperasian sistem.

2.1.1 Analisis Kebutuhan Perangkat Keras :

Perangkat keras yang digunakan dalam pembuatan sistem ini adalah Laptop (Intel Core i3-4030U CPU @ 1.90 GHz , RAM 4 GB). Sedangkan dalam pengoprasiannya dibutuhkan Laptop atau Komputer dengan RAM 4 GB.

2.1.2 Analisis Kebutuhan Perangkat Lunak :

Perangkat lunak yang digunakan dalam pembuatan sistem ini diantaranya *Windows 10*, *MySql*, *Visual Studio Code*, *PHP*, *Javascript*, dan *Framework Laravel*. *Software* yang digunakan untuk maenjalankan sistem ini menggunakan aplikasi *browser*.

2.1.3 Analisis Kebutuhan fungsional

Di dalam kebutuhan ini memiliki 2 *admin*, dan 1 *user* sebagai pengguna dalam sistem ini. *Admin* yang pertama adalah sebagai *admin* yang dapat melakukan semua CRUD (*Create*, *Read*, *Update*, *Delete*) yang ada pada sistem ini. Sedangkan *admin* petugas perbedaannya hanya tidak bisa dalam hal menambahkan *admin* petugas yang baru. Pada *User* hanya dapat melihat pembayaran yang telah dilakukan pada dirinya sendiri.

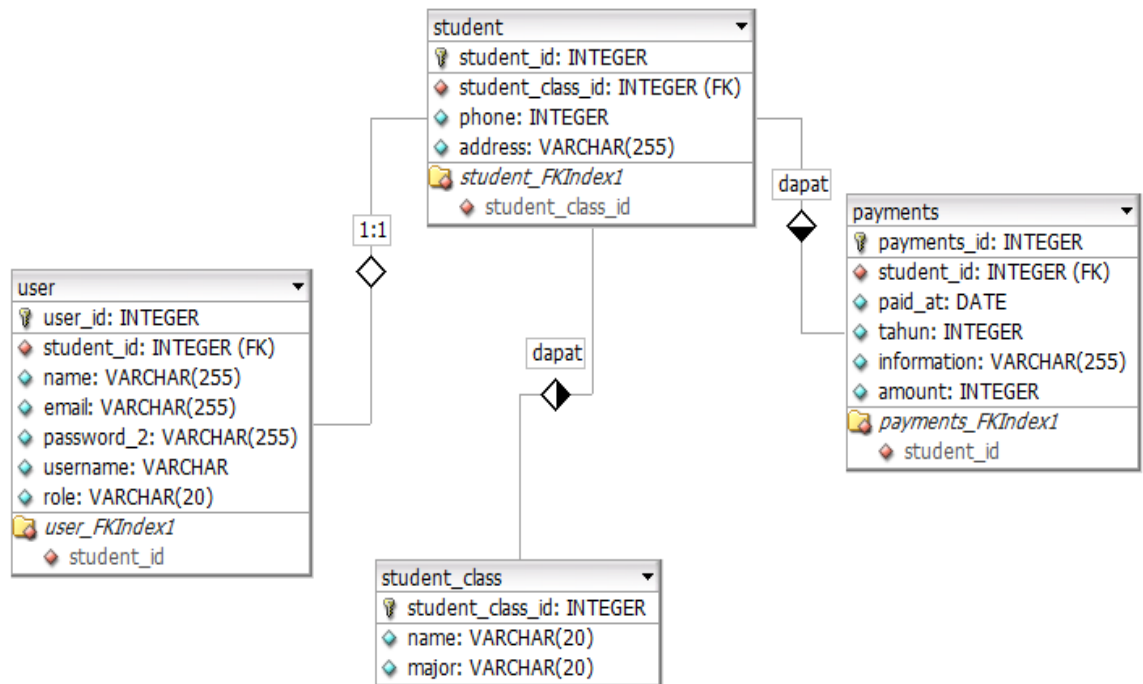
2.2 Desain atau Perancangan

Tahap ini bisa diartikan sebagai proses menerjemahkan kebutuhan yang telah dianalisis menjadi model perangkat lunak yang dapat memperkirakan kualitas sistem implementasi (Aini, Riza, Santoso, Faturahman, & Rahardja, 2020). Setelah memperoleh kebutuhan dalam sistem selanjutnya adalah mendesain sistem tersebut.

2.2.1 Perancangan Basis Data

Basisdata dalam sistem ini terdapat beberapa tabel yaitu : *user*, *payments*, *students*, *student_class*. Tabel *user* meliputi *user_id* sebagai *primary key* , *username* , *email* , *name* , *password* , *role*. Kemudian ada tabel *payments* terdapat *payments_id* sebagai *primary key*, *paid_at*, tahun, *information*, *amount*. Pada tabel *students* terdapat

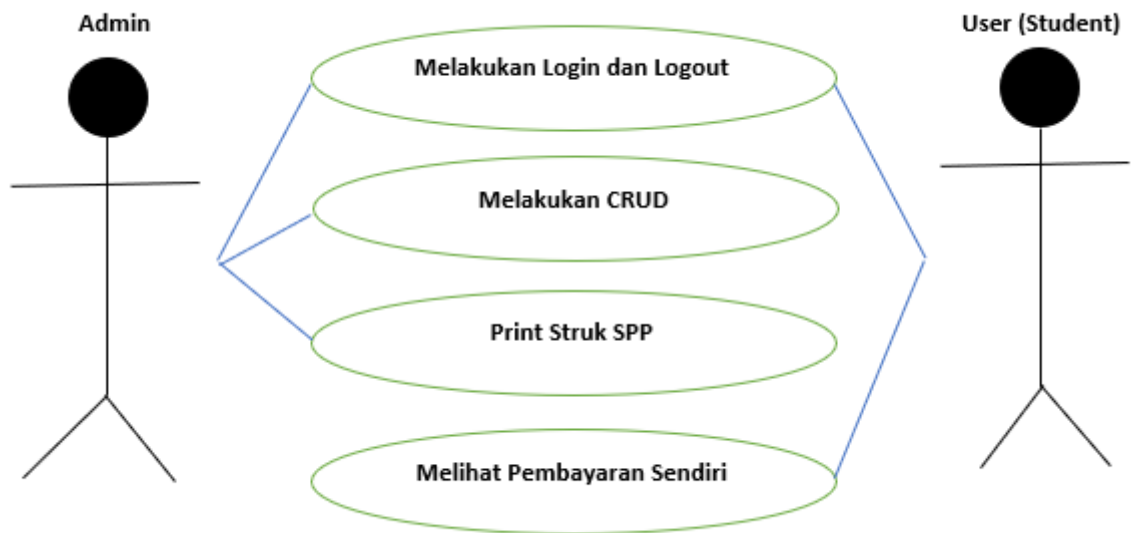
students_id, *phone*, *address*. Tabel *students_class* terdapat *students_class_id*, *name*, *major*. Perancangan tersebut dapat dilihat pada gambar 2.



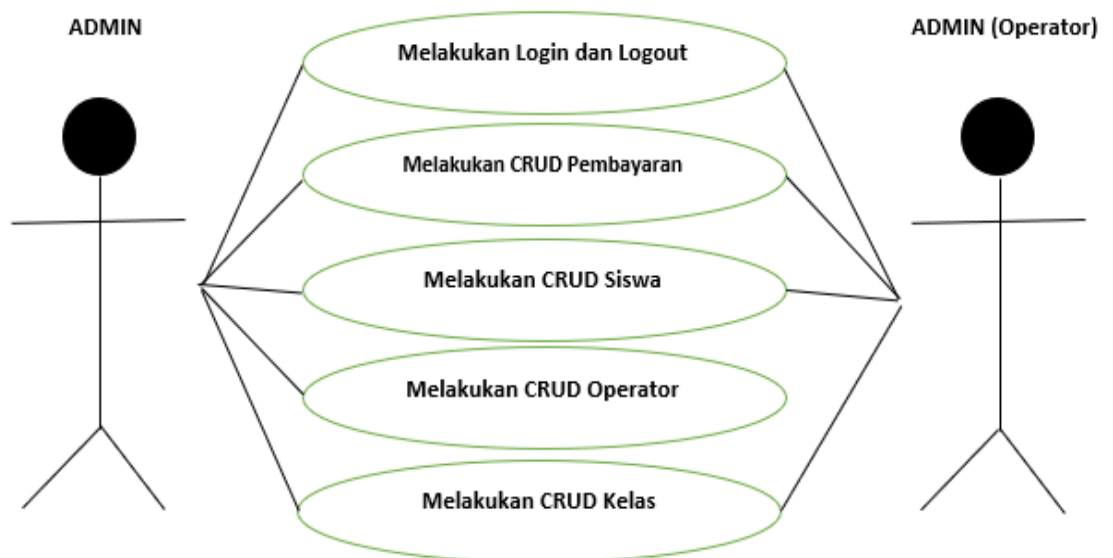
Gambar 2 Perancangan Basis Data

2.2.2 Use Case Diagram

Elemen kunci dalam model *use case* adalah aktor (entitas eksternal), dan kasus penggunaannya sendiri. Sebuah *use case* adalah unit layanan atau fungsionalitas (persyaratan), di sistem. *Use case* bukanlah program atau proses, atau fungsi secara garis besar (Ahmed & Daleel, 2020). Gambar 3 dan 4 akan menjelaskan *use case* diagram *admin* sebagai kepala keuangan dan ada *admin* sebagai operator pembayaran kemudian *user* sebagai siswa atau wali siswa yang mengecek pembayaran pada siswa itu sendiri atau anak dari wali siswa tersebut.



Gambar 3 *Use Case Diagram Admin dan User*

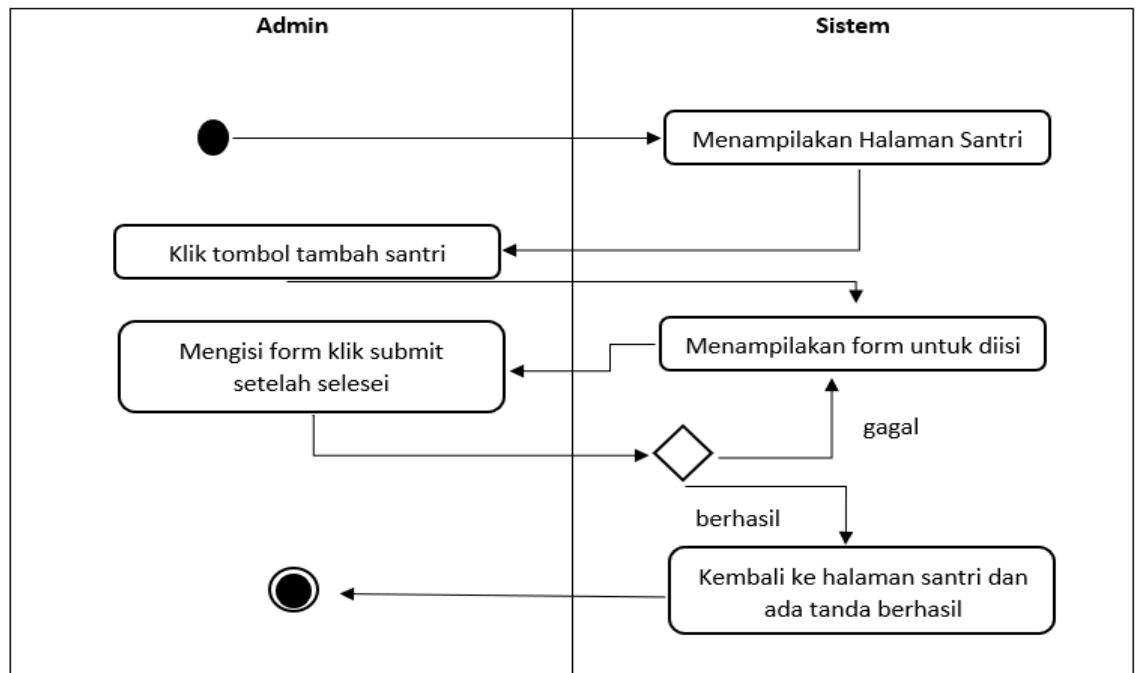


Gambar 4 *Use Case Diagram Admin dan Admin Operator*

2.2.3 Activity Diagram

- Activity Diagram* dalam menambahkan data santri. Setelah login *admin* atau operator akan masuk ke *dashboard* kemudian *admin* atau operator memilih pada tombol santri. Setelah masuk halaman santri kemudian memilih pada tombol tambah santri setelah itu muncul formulir yang harus diisi oleh *admin* atau operator. Setelah memilih *submit* maka jika berhasil akan kembali ke halaman santri dan ada

notifikasi jika berhasil ditambahkan, Jika gagal maka akan tetap dihalaman *form* dan akan mendapat notifikasi bagian yang membuat gagal. Seperti pada Gambar 5 dan 6.



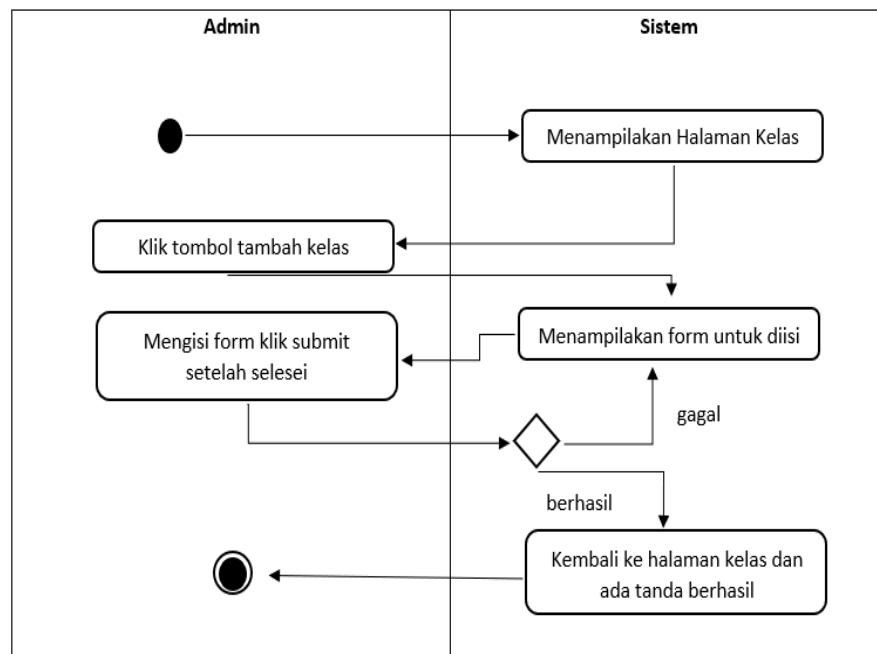
Gambar 5 Activity Diagram Menambah Santri

```

25 @endif
26
27 <form method="POST" action="{{ route('students.store') }}">
28 @csrf
29
30 <div class="form-group row">
31 <label for="nisan" class="col-md-4 col-form-label text-md-right">
32 NISN
33 </label>
34 <div class="col-md-6">
35 <input type="text" class="form-control @error('nisan') is-invalid @enderror" id="nisan" name="nisan" value="{{ old('nisan') }}" required>
36
37 @error('nisan')
38 <span class="invalid-feedback" role="alert">
39 <strong>{{ $message }}</strong>
40 </span>
41 @enderror
42 </div>
43 </div>
44
45 <hr>
46
47 <div class="form-group row">
48 <label for="name" class="col-md-4 col-form-label text-md-right">
49 Nama
50 </label>
51 <div class="col-md-6">
52 <input type="text" class="form-control @error('name') is-invalid @enderror" id="name" name="name" value="{{ old('name') }}" required>
53
54 @error('name')
55 <span class="invalid-feedback" role="alert">
56 <strong>{{ $message }}</strong>
57 </span>
  
```

Gambar 6. Potongan Kode Halaman *Form* Tambah Santri

- b. *Activity Diagram* dalam menambahkan data kelas. Setelah login *admin* atau operator akan masuk ke *dashboard* kemudian *admin* atau operator memilih pada tombol kelas. Setelah masuk halaman kelas kemudian memilih pada tombol tambah santri setelah itu muncul *form* yang harus diisi oleh *admin* atau operator. Setelah memilih *submit* maka jika berhasil akan kembali ke halaman kelas dan ada notifikasi jika berhasil ditambahkan, Jika gagal maka akan tetap dihalaman *form* dan akan mendapat notifikasi bagian yang membuat gagal. Seperti pada Gambar 7 dan 8.



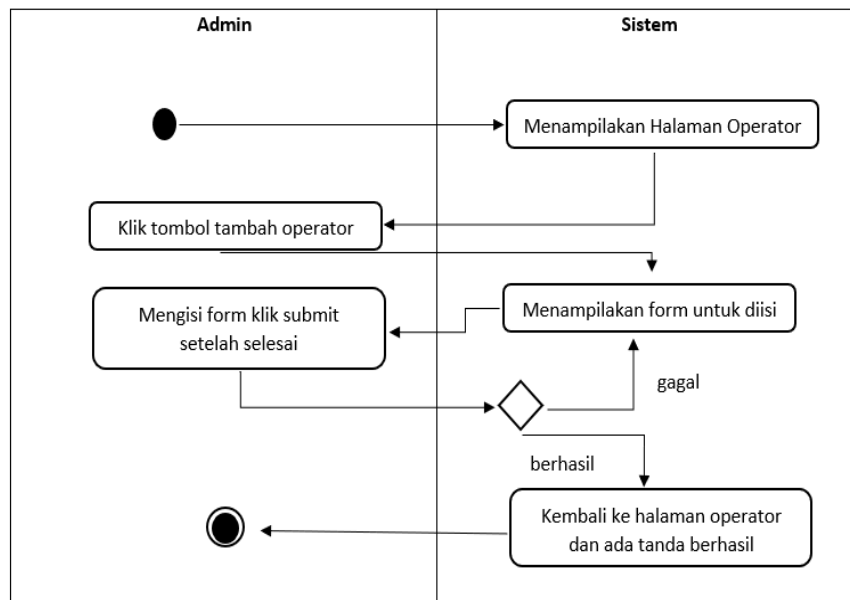
Gambar 7. Activity Diagram Menambah Kelas

```

25 @endif
26
27 <form method="POST" action="{{ route('student-classes.store') }}">
28 @csrf
29
30 <div class="form-group row">
31 <label for="name" class="col-md-4 col-form-label text-md-right">
32 Nama Kelas
33 </label>
34 <div class="col-md-8">
35 <input type="text" class="form-control @error('name') is-invalid @enderror" id="name" name="name" value="{{ old('name') }}" required="">
36
37 @error('name')
38 <span class="invalid-feedback" role="alert">
39 <strong>{{ $message }}</strong>
40 </span>
41 @enderror
42 </div>
43 </div>
44
45 <div class="form-group row">
46 <label for="major" class="col-md-4 col-form-label text-md-right">
47 Major
48 </label>
49 <div class="col-md-8">
50 <input type="text" class="form-control @error('major') is-invalid @enderror" id="major" name="major" value="{{ old('major') }}" required="">
51
52 @error('major')
53 <span class="invalid-feedback" role="alert">
54 <strong>{{ $message }}</strong>
55 </span>
56 @enderror
57 </div>
  
```

Gamabar 8. Potongan Kode Halaman *Form* Tambah Kelas

- c. *Activity Diagram* dalam menambahkan data operator. Setelah login *admin* akan masuk ke *dashboard* kemudian *admin* memilih pada tombol operator. Setelah masuk halaman kelas kemudian memilih pada tombol tambah operator setelah itu muncul *form* yang harus diisi oleh *admin*. Setelah memilih *submit* maka jika berhasil akan kembali ke halaman operator dan ada notifikasi jika berhasil ditambahkan, Jika gagal maka akan tetap dihalaman *form* dan akan mendapat notifikasi bagian yang membuat gagal. Seperti pada Gambar 9 dan 10.



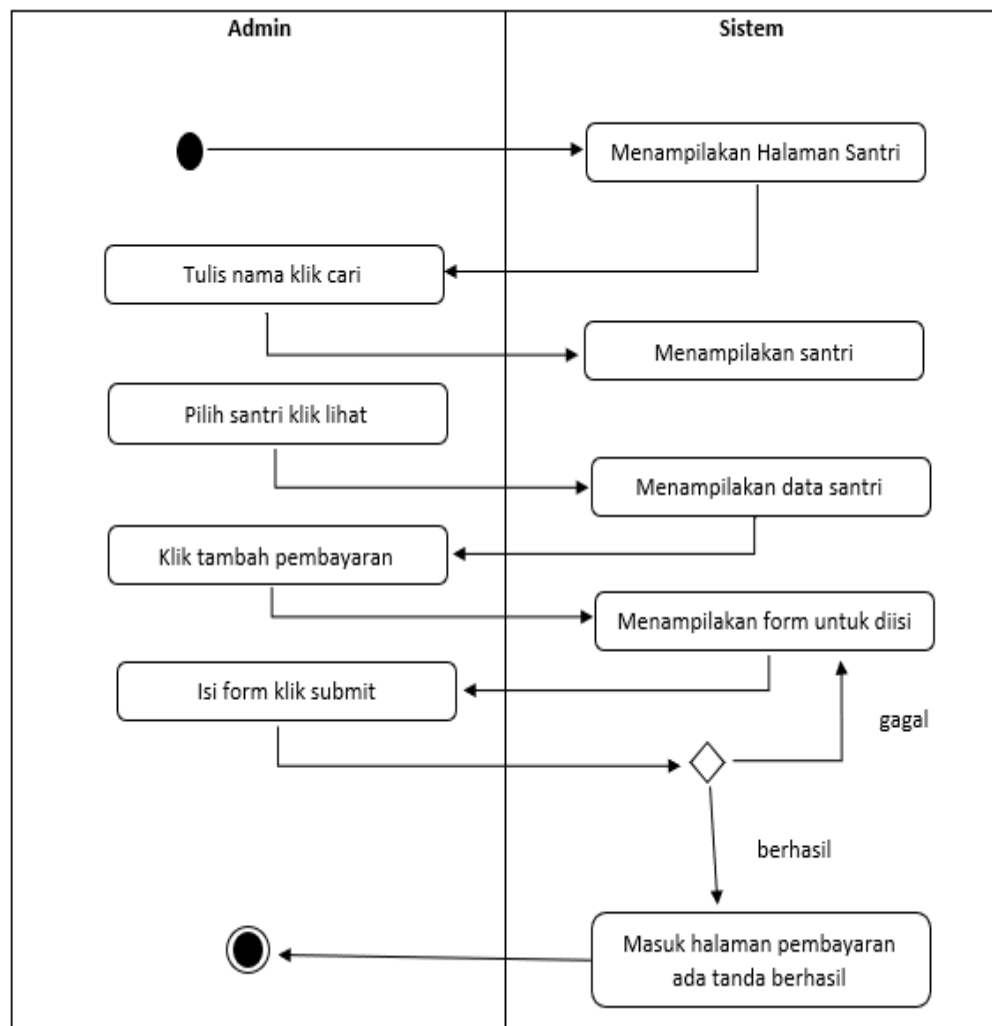
Gambar 9. *Activity Diagram* menambahkan operator oleh *admin*

```

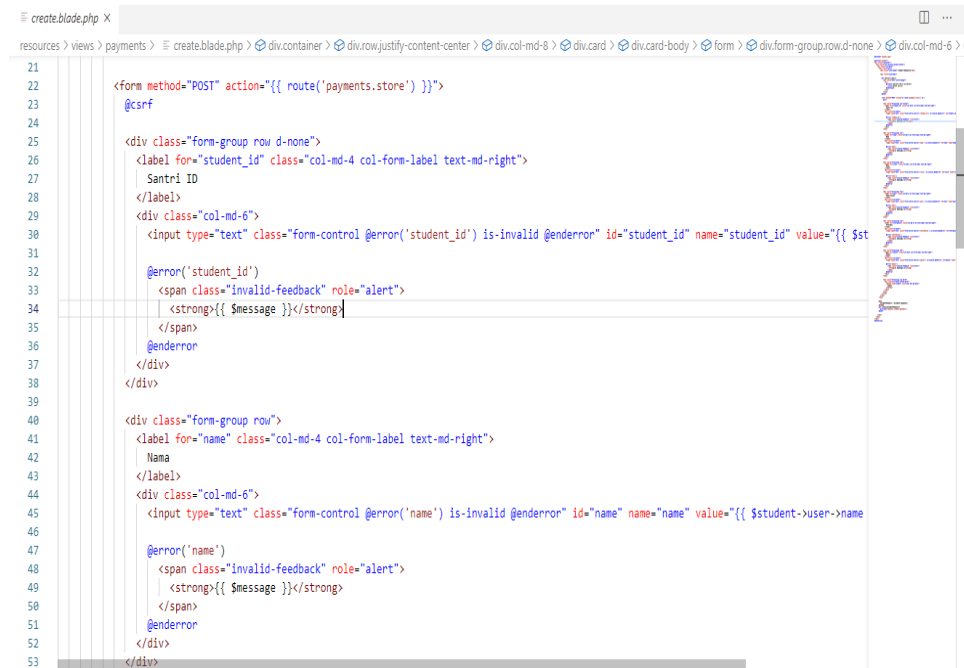
22 <form method="POST" action="{ route('operators.store') }">
23   @csrf
24
25   <div class="form-group row">
26     <label for="username" class="col-md-4 col-form-label text-md-right">
27       Username
28     </label>
29     <div class="col-md-8">
30       <input type="text" class="form-control @error('username') is-invalid @enderror" id="username" name="username" value="{{ old('user
31
32       @error('username')
33         <span class="invalid-feedback" role="alert">
34           <strong>{{ $message }}</strong>
35         </span>
36       @enderror
37     </div>
38   </div>
39
40   <div class="form-group row">
41     <label for="name" class="col-md-4 col-form-label text-md-right">
42       Nama
43     </label>
44     <div class="col-md-8">
45       <input type="text" class="form-control @error('name') is-invalid @enderror" id="name" name="name" value="{{ old('name') }}" requi
46
47       @error('name')
48         <span class="invalid-feedback" role="alert">
49           <strong>{{ $message }}</strong>
50         </span>
51       @enderror
52     </div>
53   </div>
54
  
```

Gambar 10. Potongan Kode Halaman *Form* Tambah Operator

- d. *Activity Diagram* dalam menambahkan pembayaran. Setelah *admin* atau operator login dan masuk ke *dashboard* masing-masing kemudian memilih ke halaman santri kemudian cari nama santri yang akan dilakukan pembayaran. Pilih lihat pada santri yang akan dilakukan pembayaran setelah itu pilih pada tombol tambah pembayaran maka akan muncul *form* untuk pembayaran. Setelah diisi dengan lengkap pilih *submit* maka jika berhasil akan masuk pada halaman pembayaran dan ada pemberitahuan pembayaran berhasil ditambahkan, jika gagal akan tetap pada *form*. Seperti pada Gambar 11 dan 12.



Gambar 11. *Activity Diagram* Menambah Pembayaran



Gambar 12. Potongan Kode Halaman *Form* Tambah Pembayaran

2.3 Pengkodean

Menurut Anggoro dan Syahputri (2020) Pengkodean merupakan tahap pembuatan kode program dengan memanfaatkan bahasa pemrograman HTML, PHP, dan CSS. Pada tahap ini sistem informasi yang dibuat menggunakan bahasa pemrograman HTML, PHP, dan CSS. Penggunaan bahasa pemrograman tersebut dikarenakan bahasa yang mudah dipelajari dan dapat dijalankan di setiap web *browser*.

2.4 Pengujian

Black-box adalah pengujian yang dilakukan dalam pembuatan program ini. Pengujian dilakukan untuk mendeteksi kesalahan atau cacat pada penelitian sistem aplikasi. Pengujian bertujuan untuk mengevaluasi kerangka aplikasi yang memenuhi tujuan desain sistem aplikasi (Rambe, Pane, Irmayani, Nasution, & Munthe, 2020).

2.5 Pemeliharaan

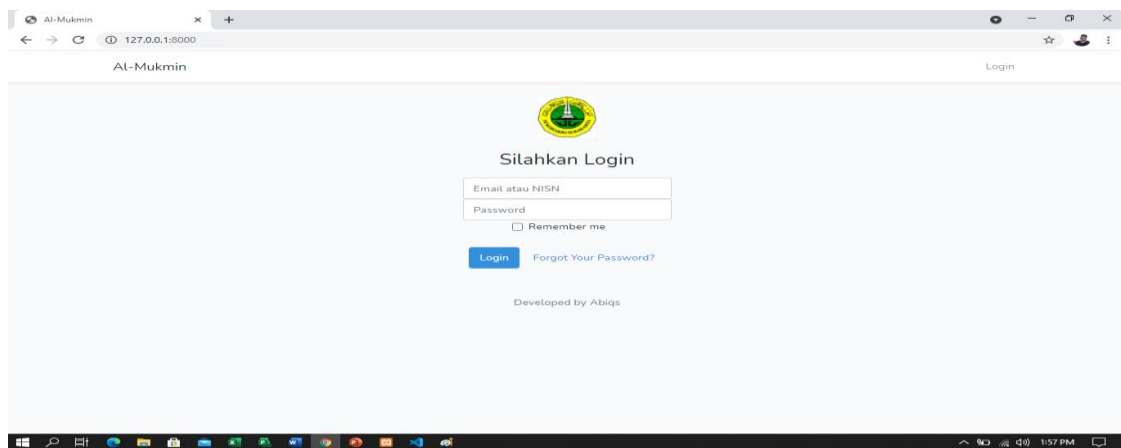
Perawatan dilakukan agar sistem bisa berjalan sesuai rancangan awal dalam jangka waktu yang panjang. Selain itu, perawatan dilakukan agar bisa menyesuaikan kebutuhan *user* apabila terjadi perubahan atau penambahan kebutuhan (Putri & Supriyono, 2019). Pemeliharaan sistem dibutuhkan agar sistem dapat berjalan sesuai dengan tugasnya sehingga tidak memperlambat dalam pengerjaan.

3. HASIL DAN PEMBAHASAN

Hasil dari penelitian yang didapat yaitu menghasilkan suatu sistem informasi pembayaran bagi Pondok Pesantren Islam Al-Mukmin yang berbasis website. Sistem ini membantu bagian pembayaran atau *syahriah* dalam pembayaran. Sistem ini mempunyai beberapa tampilan baik untuk *admin* ataupun untuk *user*. Berikut ini adalah hasil dari pembahasan.

3.1 Halaman Login

Halaman Login adalah halaman dimana semua pengguna melakukan verifikasi agar bisa masuk ke halaman *dashboard*. Berikut adalah halaman login pada gambar 13.

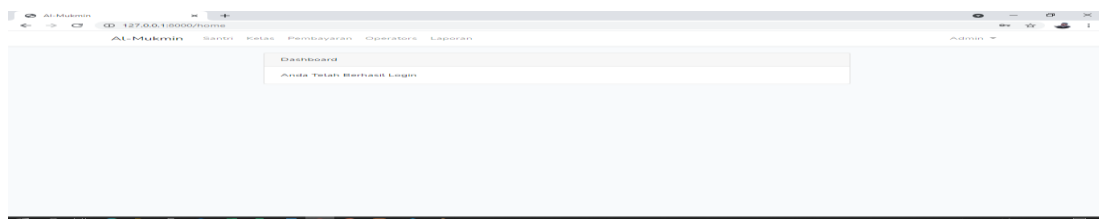


Gambar 13. Tampilan Halaman Login

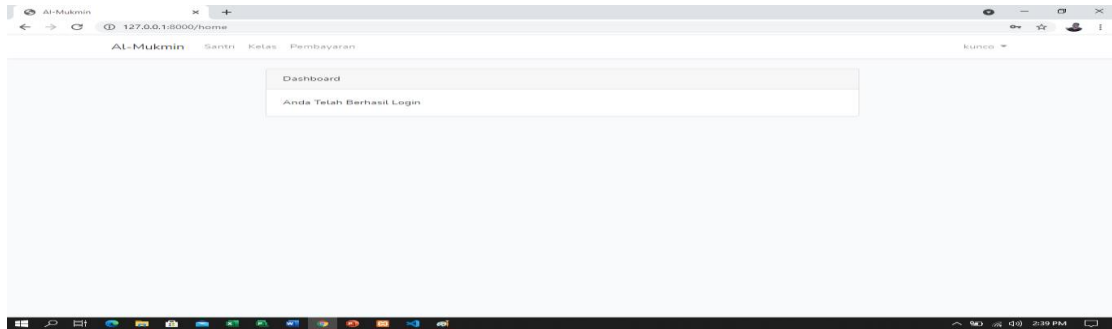
3.2 Halaman *Dashboard*

Halaman *dashboard* adalah halaman awal pengguna setelah melakukan verifikasi atau login. Setiap halaman *dashboard* masing-masing pengguna sistem ini berbeda-beda seorang *admin* akan dapat menu awal pada *navbar* yaitu unruk masuk ke halaman santri, kelas, operator, pembayaran, dan halaman laporan. Untuk seorang operator akan mendapatkan menu santri, kelas, pembayaran, laporan. Pada halaman *dashboard* seorang *user* hanya akan masuk ke pembayaran dirinya sendiri.

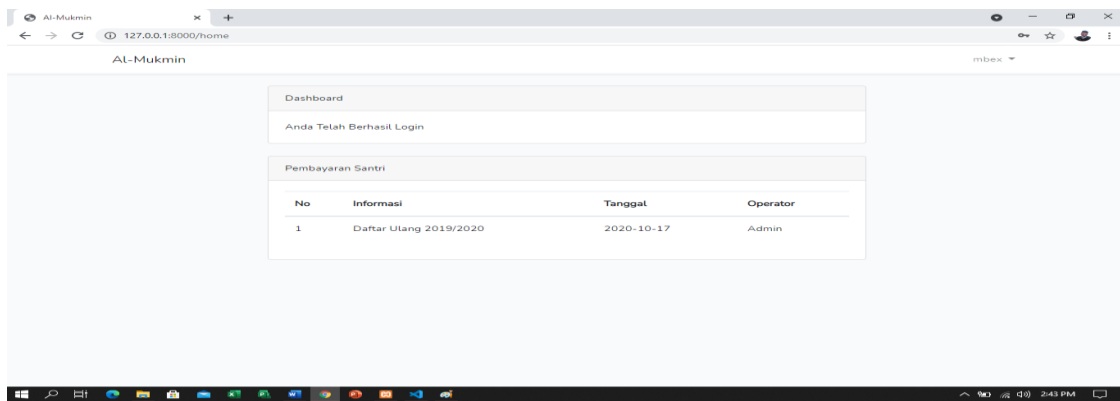
Gambar *dashboard* pada masing-masih pengguna ditunjukan pada gambar 14, 15, dan 16 berikut.



Gambar 14. Halaman *Dashboard Admin*



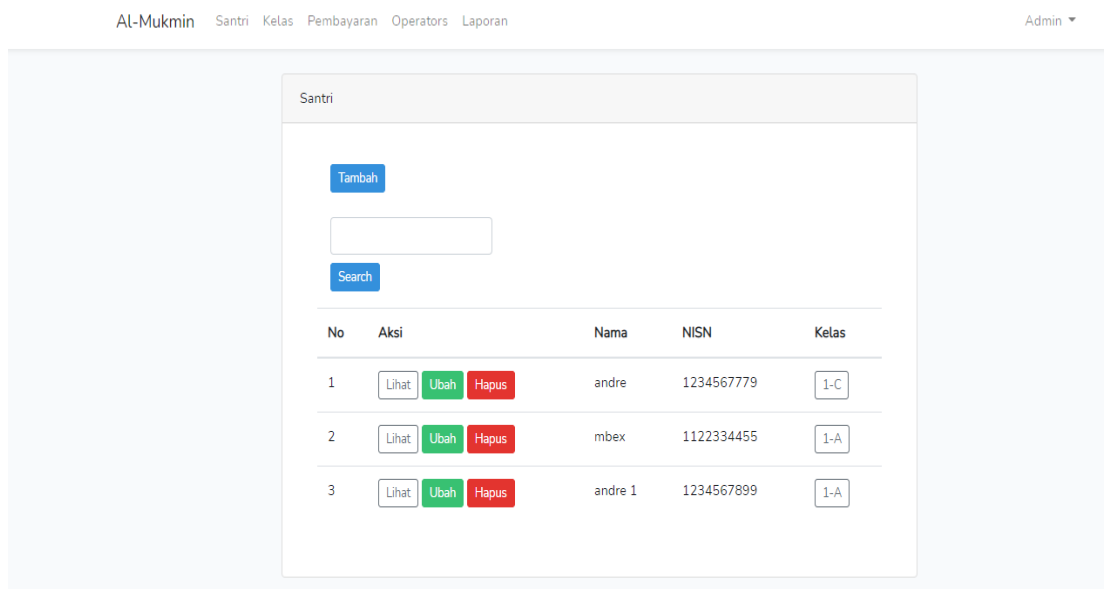
Gambar 15. Halaman *Dashboard* Operator



Gambar 16. Halaman *Dashboard* User

3.3 Halaman Santri

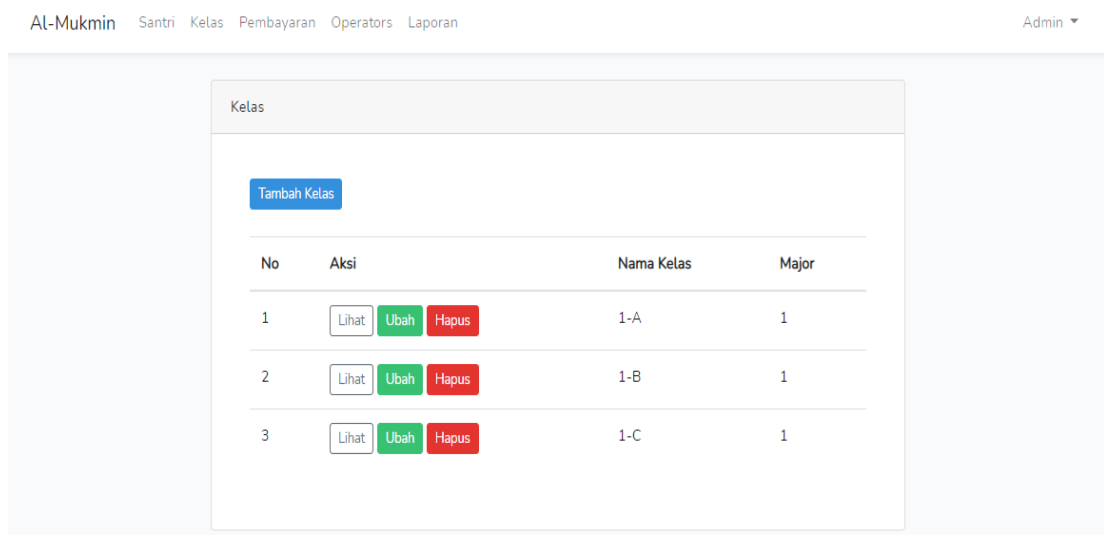
Halaman santri terdapat beberapa menu yaitu tambah yang berguna untuk menambah santri, kemudian terdapat menu untuk mencari santri dengan nama, dan tabel santri yang berisikan nama, NISN, kelas, dan terdapat beberapa menu untuk mengubah, menghapus dan melihat data santri tersebut. Gambar 17 menunjukan halaman santri.



Gambar 17. Halaman Santri

3.4 Halaman Kelas

Halaman kelas terdapat menu tambah yang berguna untuk menambah kelas, dan tabel kelas yang berisikan nama sebagai nama kelas, major sebagai dasar kelas tersebut, dan terdapat beberapa menu untuk mengubah, menghapus dan melihat data kelas tersebut. Gambar 18 menunjukan halaman kelas.



Gambar 18. Halaman Kelas

3.5 Halaman Pembayaran

Halaman pembayaran berisikan tabel yang memberitahukan pembayaran yang telah dilakukan dalam tabel tersebut berisikan no, nama dari pembayarnya, kelas dari si pembayar, informasi yang memberitahukan guna pembayaran itu dilakukan, tanggal

sebagai kapan pembayaran tersebut dilakukan, operator yaitu siapa yang membuat pembayaran tersebut, dan aksi yang didalamnya terdapat *print* untuk mencetak struk pembayaran dan hapus untuk menghapus pembayaran. Gambar 19 menunjukkan halaman pembayaran.

Al-Mukmin Santri Kelas Pembayaran Operators Laporan Admin ▾

Pembayaran						
No	Nama	Kelas	Informasi	Tanggal	Operator	Aksi
1	andre	1-C	January 2019/2020	2020-10-21	Admin	Print Hapus
2	mbex	1-A	Daftar Ulang 2019/2020	2020-10-17	Admin	Print Hapus

Gambar 19. Halaman Pembayaran

3.5 Halaman Operator

Halaman operator adalah halaman yang hanya dapat dibuka atau diakses oleh *admin* saja. Halaman operator terdapat tabel operator yang berisikan no, aksi yang berisikan beberapa menu diantaranya hapus yang berguna untuk menghapus, Ubah untuk mengubah, Lihat untuk melihat data dari operator tersebut, dalam tabel tersebut terdapat nama untuk nama operator itu sendiri dan *username* sebagai identitas khusus operator. Gambar 20 menunjukkan halaman operator.

Al-Mukmin Santri Kelas Pembayaran Operators Laporan Admin ▾

Operators				
Tambah Operator				
No	Aksi	Nama	Username	
1	Lihat Ubah Hapus	kunco	kunc	

Gambar 20. Halaman Operator

3.6 Halaman Laporan

Halaman laporan adalah halaman untuk mencetak laporan pertanggal yang dibutuhkan. Halaman laporan ini digunakan untuk mencetak laporan yang akan diserahkan kepada

yayasan. Pada halaman laporan terdapat tanggal awal yaitu tanggal awal yang akan dicetak dan tanggal akhir sebagai tanggal terakhir laporan yang dibutuhkan, seperti gambar 21 dibawah.

Gambar 21. Halaman Laporan

3.7 Pengujian *Black-Box*

Menurut Hidayat dan Putri (2019) *Black-Box Testing* merupakan pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak, *tester* dapat mendefinisikan kumpulan kondisi input dan melakukan pengetesan pada spesifikasi fungsional program. Hasil pengujian *Black-Box* dilakukan pada saat melakukan testing oleh bagian keuangan atau *syahriah* Pondok Pesantren Al Mukmin yang berjumlah 4 orang. Agar dapat mengetahui adanya kesalahan pada sistem maka setiap orang melakukan pengujian beberapa kali. *Testing* dilakukan oleh bagian keuangan atau *syahriah* karena yang akan menggunakan sistem yang dibuat ini. Setelah *testing* dilakukan, mendapatkan kesimpulan yang ditunjukkan pada tabel 1.

Tabel 1. Hasil Pengujian *Black-Box* dengan Skenario dan Harapan

No	Uji	Skenario	Harapan	Hasil
1	Login	Mengisi <i>Username</i> /Email dan <i>Password</i> Benar	Sistem masuk Dashboard	Valid
		Mengisi <i>Username</i> /Email dan <i>Password</i> Salah	Sistem tetap dihalaman Login dan pemberitahuan salah	Valid
2	Logout	Menekan Tombol Logout	Kembali Ke halaman login	Valid
3	Menu	Menekan Menu Santri	Masuk Halaman Santri dan	Valid

	Santri		Menampilkan Tabel Santri	
		<i>Admin</i> Dan Operator dapat Melakukan CRUD pada santri	Data Santri berhasil Ditambah, Hapus, Lihat, dan diubah	Valid
4	Menu Kelas	Menekan Menu Kelas	Masuk Halaman Kelas dan Menampilkan Tabel Kelas	Valid
		<i>Admin</i> Dan Operator dapat Melakukan CRUD pada Kelas	Data Kelas berhasil Ditambah, Hapus, Lihat, dan diubah	Valid
5	Menu Operator	Menekan Menu Operator	Masuk Halaman Operator dan Menampilkan Tabel Operator	Valid
		<i>Admin</i> dapat Melakukan CRUD pada Operator	Data Operator berhasil Ditambah, Hapus, Lihat, dan Ubah	Valid
6	Menu Pembayaran	Menekan Menu Pembayaran	Masuk Halaman Pembayaran dan Menampilkan Tabel Pembayaran	Valid
		<i>Admin</i> dan Operator Dapat Menghapus dan Mencetak Pembayaran	Pembayaran Berhasil di hapus dan masuk View cetak struk pembayaran	Valid
6	Menu Pembayaran	<i>Admin</i> dan operator mencari siswa/santri yang akan melakukan pembayaran kemudian membuka detail siswa dan menekan tambah pembayaran dan mengisi <i>form</i>	Jika berhasil sistem akan masuk ke Halaman pembayaran dan terdapat data pembayaran yang baru	Valid
			Jika gagal menambah pembayaran maka akan tetap pada halaman <i>form</i> pembayaran dan ada	Valid

			pemberitahuan letak kesalahan	
7	Menu Laporan	<i>Admin</i> dan Operator Menekan Menu Laporan	Masuk ke Halaman Laporan	Valid
		<i>Admin</i> dan Operator Mencetak laporan dengan memilih Tanggal awal dan tanggal akhir laporan yang akan dicetak	Masuk ke halaman cetak dan muncul tabel beserta isinya yang akan dicetak	Valid

4. PENUTUP

Sistem Informasi Pembayaran telah berjalan sesuai dengan apa yang diharapkan. Sistem ini bertujuan untuk memudahkan bagian pembayaran dalam pembukuan yang dilakukan. Berdasarkan pengujian *Black-Box*, semua fitur yang telah dibuat dapat berjalan sesuai dengan apa yang diharapkan. Pembuatan sistem ini adalah untuk mempermudah bagian keuangan dalam melakukan pembayaran, mencetak struk, dan membuat laporan. Agar lebih efektif dan efisien melakukan pengerjaan dalam proses pembayaran santri. Pada pengembangan sistem ini, selanjutnya diharapkan dapat menambah beberapa fitur yang lebih menarik.

DAFTAR PUSTAKA

- Ahmed, I. A. O., & Daleel, M. E. E. (2020). Automated Use Case Diagram Generation with Nonfunctional Requirements using Neural Network. *International Journal of Applied Information Systems (IJAIS)*. 12 (34).
- Aini, Q., Riza, B. S., Santoso, N. P.L., Faturahman, A., & Rahardja, U.(2020). Digitalization of Smart Student Assessment Quality in Era 4.0. *International Journal of Advanced Trends in Computer Science and Engineering*. 9 (1.2).257-265.
- Anggoro, D. A., & Arina N. S. (2020). Penerapan Sistem Informasi Penjualan Dengan Platform E-Commerce Pada Perusahaan Daerah Apotek Sari Husada Demak. *Sintech Jurnal*.3 (1).59-70.
- Bhavsar, K., Shah,V., & Gopalan, S.(2020). Scrumbanfall: An Agile Integration of Scrum and Kanban with Waterfall in Software Engineering. *International Journal of Innovative Technology and Exploring Engineering*. 9 (4). 2075-2084.
- David, K. M., Tanui, E., dan Oruta, F.(2019). Determining the extent of ICT implementation and use in financial management of Secondary Schools in

- Kajiado County, Kenya. *Editon Consortium Journal of Business and Management Studies*. 1 (1).18-27.
- Susanto, E.(2018).Sistem Informasi Pembayaran SPP Berbasis Web di MTS Baiturrahman Beringin Taluk.*Jurnal Perencanaan,Sains,Teknologi, dan Komputer*.1(2).141-146.
- Gharajeh, M. S.(2019). Waterative Model: an Integration of the Waterfall and Iterative Software Development Paradigms.*Database System Journal*.10 (8).75-81.
- Hidayat, T., & Putri, H. D.(2019). Pengujian Portal Mahasiswa pada Sistem Informasi Akademik (SINA) menggunakan Black Box Testing dengan Metode Equivalence Partitioning dan Boundary Value Analysis. *JUTIS*. 7 (1).83-92.
- Putri, D. P. M., & Supriyono, H.(2019). Rancang Bangun Sistem Presensi Berbasis QR Code Menggunakan Framework Codeigniter (Studi Kasus Kehadiran Asisten Praktikum).*Jurnal insypro*. 4 (1).1-9.
- Rambe , B. H., Pane, R., Irmayani. D., Nasution. M., & Munthe I. R.(2020). UML Modeling and Black Box Testing Methods in the School Payment Information System. *Jurnal Mantik*. 4 (3).1634-1640.